By -
Raluca Mihaela Andrei,
Mike Pan
and Monica Zoppè

Raluca Mihaela Andrei1,2, Mike Pan1,* and Monica Zoppè1§

1 Scientific Visualization Unit, Institute of Clinical Physiology, CNR of Italy, Area della Ricerca, Pisa, Italy
2 Scuola Normale Superiore, Pisa, Italy
* Present address: University of British Columbia, Vancouver, Canada
§ Corresponding author

Introduction

Biologists know that, if the information of life is stored and transmitted through nucleic acids (DNA and RNA), the processes that do the actual work are most of the times proteins. These are active in all aspects of life, and in the latest years we are starting to get a glimpse of how they work. Proteins are machines composed of amino acids, which are in turn small groups of atoms arranged in specific ways[1]. Scientists are obtaining more and more information on the 3D arrangement of such atoms, and are starting to understand their activity through motion.

On the basis of information obtained by experiments of nuclear magnetic resonance (NMR), 3D visualization tools provided by BioBlender allow biologists to build a reasonable sequence of movement for proteins. It also includes a dedicated visual code to represent important features of their surface (Electric and lipophilic potential) on the protein itself, using photo realistic rendering and special effects.

BioBlender is a software extension of Blender 2.5[2], an interface for biological visualization that allows the user to import and interactively view and manipulate proteins. It was developed and is maintained by the Scientific Visualization Unit of the CNR of Italy in Pisa, with the help and contribution of several members of the Blender community. Material, scenes, publications and other relevant information can be found at www.BioBlender.net and/or www.scivis.ifc.cnr.it.

BioBlender for Windows is available from www.bioblender.net (on Linux machines it can be used with Wine). Because of its specialized nature, it requires the installation of PyMOL[3,4] , Python 2.6 [5] and NumPy[6] , which are all provided in Installer folder from the downloaded package.

### Using BioBlender to build an animation

To start BioBlender, simply go to the Bin folder and launch blender.exe, then open the **template.blend** scene (stored in BioBlender folder).

Notice that the template file not only has an optimized user-interface layout for biologists, but the template scene also contains lights, camera and world settings that are ideal for visualizing molecules. This setup ensures that researchers who are not familiar with the 3D software can still effectively use BioBlender. Each interface element (buttons, sliders, toggles) has help text associated with it. By placing the mouse over them a pop-up text describes the function. Errors and progresses are displayed in the console. Critical errors will appear in the main BioBlender as a pop-up under the mouse. The atoms size is of order of Ångström (Å), therefore the scale used is 1 Blender Unit = 1 Å.

This tutorial assumes that you already have BioBlender downloaded on your computer, with the required programs installed.

1. Select and import a .pdb file

PDB files contain a description of one or multiple conformations (positions) of a single molecule. Different conformations of the same protein are listed in one NMR file and are called MODEL 1, MODEL 2 etc.

Raluca Mihaela Andrei[1,2], Mike Pan[1,]* and Monica Zoppè[1]§

1 Scientific Visualization Unit, Institute of Clinical Physiology, CNR of Italy, Area della Ricerca, Pisa, Italy
2 Scuola Normale Superiore, Pisa, Italy
* Present address: University of British Columbia, Vancouver, Canada
§ Corresponding author

In the **BioBlender Select PDB** File panel:

- Select the .pdb file by browsing from your data (**1** in figure). The file included in sampleData folder contains the 25 models of Calmodulin [7]. Alternatively, simply type the 4-letter code for the .pdb file to be fetched from www.pdb.org [8] (make sure to pick an NMR file);

- Change the name of the protein (by default it is named "protein0") in the field on the right (**2** in figure). Naming the proteins is just a good habit that will help keeping the scene organized. Once a file is selected, the number of models and the chains are detected and shown in the BioBlender Import field (**3** in figure);

- Choose 2 models to import in the scene (by default all models are listed) typing their number separated by comma;

- In the **Keyframe Interval** slider (**4** in figure) set the number of frames between the protein conformations (Min 1, Max 200).

A list of options are available to be considered before importing the protein in the Blender scene (**5** in figure):

Verbose: enable to display in the console extra information for debugging;

SpaceFill: enable or disable to display the atoms with Van der Waals or covalent radii in the 3D scene, respectively;

Hydrogen: enable to import Hydrogens if they are present in the .pdb file. This option makes importing much slower and it is important only for visualization. If the .pdb file does not contain Hydrogens (or if you chose not to import them), they will be added during the Electrostatic Potential calculation using external software;

Make Bonds: enable it to have atoms connected by chemical bonds. Despite being time consuming this operation is very important in motion calculation;

High quality: displays high-quality atom and surface geometries; slow when enabled;

Single User: enable to use shared mesh for atoms in Game Engine; slow when enabled;

Upload Errors: enable to send us automatically and anonymously an email with the errors you generate. This makes us aware of the problems that arise and help us fix them.

Finally, press **Import PDB** button to import the protein to the 3D scene of Blender. Blender displays the protein in motion (by linear interpolation between atoms in the conformations; Esc to stop the animation).

2. Visualization in the 3D viewport

Once imported, the protein is displayed with all atoms, Hydrogens included (if the Hydrogens check-box was enabled). The first 4 buttons in the **BioBlender View** enable different views: only alpha Carbons, main chain (N, CA, C), main chain and side chains (no H), or all atoms.

If the **Surface** display mode is selected, BioBlender will compute the surface of the protein by invoking PyMOL software, an external application. It uses the **Solvent Radius** set by the user and returns the Connolly mesh [9], displayed on the BioBlender 3D view. The default radius (1.4 Å) is the standard probe sphere, equivalent to water molecules.

To check the appearance of surface calculated with different solvent radii, change the solvent radius value and press refresh button. The current surface is deleted and a new one is created.

When atoms are displayed, by selecting one atom in the 3D display, the protein information of the selected atom is printed in the area outlined below; in the 3D view the selection will extend to the other atoms of he corresponding aminoacid.

3. Protein motion using the physic engine

To calculate the transition of the protein between the 2 conformations the Blender Physics Engine is used. Press **Run in Game Engine** button to see the transition. Press Esc to leave GE and then 0 on Numerical Board to see from the camera point of view.

Hit **Run in Game Engine** button again for an interactively view. When inside the Game Engine, the mouse controls the rotation of the protein, allowing to inspect the protein from all angles. The also applies an ambient occlusion filter to the scene, giving the viewer a much better sense of depth.

Set the **Collision** mode to one of the following states: 0, 1 or 2. When set to 0 the transition between the conformations is done using linear interpolation; the atoms will simply move from one position to the other. When set to 1 the collisions between atoms are considered, resulting in a more physico-chemical accurate simulation[10].

When set to 2, the newly evaluated movement will be record to F-Curves. Go to the Timeline panel on Blender and see that the new conformations are recorded at different time (200 frames away from the last model imported) as shown in the figure below; in this way both sets of transitions are available for comparison. These conformations can be exported as described later in section 6.
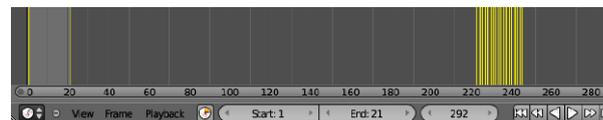
Raluca Mihaela Andrei[1,2], Mike Pan[1],* and Monica Zoppè[1]§

1 Scientific Visualization Unit, Institute of Clinical Physiology, CNR of Italy, Area della Ricerca, Pisa, Italy
2 Scuola Normale Superiore, Pisa, Italy
* Present address: University of British Columbia, Vancouver, Canada
§ Corresponding author

Raluca Mihaela Andrei[1,2], Mike Pan[1],* and Monica Zoppè[1]§

[1] Scientific Visualization Unit, Institute of Clinical Physiology, CNR of Italy, Area della Ricerca, Pisa, Italy
[2] Scuola Normale Superiore, Pisa, Italy
* Present address: University of British Columbia, Vancouver, Canada
§ Corresponding author

4. Molecular Lipophilic Potential Visualization

This visualization method is a novel way to see the MLP values of a protein onto the surface. Normally this is a relatively time consuming and tedious process involving running different programs from the command line, but BioBlender simplifies the entire process by allowing the user to do everything under one unified interface.

In **BioBlender MLP Visualization** section:

- Choose a **Formula** (**1** in figure; Testa formula [11] is set by default);

- Set the Grid Spacing (2 in figure; expressed in Å, lower is more accurate but slower) for MLP calculation;

- Press **Show MLP on Surface**. It may take some time as the MLP is calculated in every point of the grid in the protein space, then mapped on the surface of the protein and finally visualized as levels of grey (light areas for hydrophobic and dark areas for hydrophilic [12]).

A typical protein has varying degrees of lipophilicity distributed on its surface, as shown here for CaM.

Use **Contrast** and **Brightness** sliders to enhance the MLP representation of your protein. Once you are satisfied with the grey-levels visualization hit **Render MLP to Surface** button for the photorealistic render. This process is also time consuming and it always refers to last changes in the MLP grey-levels visualization. When the calculation is done (the button is released) press F12 on your keyboard.
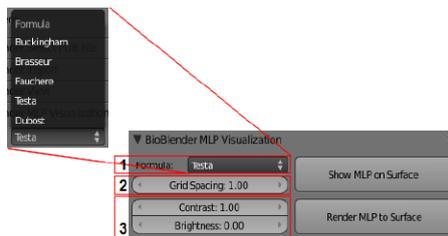
Note: This is the MLP representation using our novel code: a range of visual features that goes from shiny-smooth surfaces for hydrophobic areas to dull-rough surfaces for hydrophilic ones. The levels of grey are baked as image texture that is mapped on specular of the material. A second image is created by adding noise to the first one and map it on bump. The light areas become shiny and smooth while the dark ones dull and rough as shown in the figure.
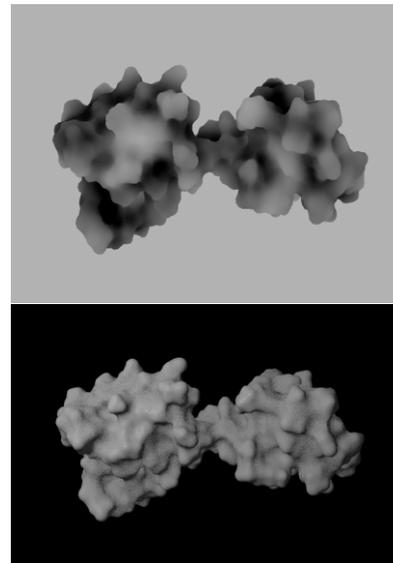
Press Esc to go back to the Blender scene.

5. Electrostatic Potential Visualization

EP is represented as a series of particles flowing along field lines calculated according to the potential field due to the charges on the protein surface. For this reason, it is necessary to perform a series of steps (as described in [12]), and to decide the physical parameters to be used in the calculation (2 in the figure).

In **BioBlender EP Visualization section**:

- Choose a **ForceField** (**1** in figure; amber force field is set by default);

- Set the parameters for EP computation, using the options shown in the figure below:

- **Ion concentration** – 0.15 Molar is the default, physiological value;

- **Grid Spacing** – in Å, lower is more accurate but slower;

- **Minimum Potential** – the minimum value for which the field lines are calculated – the default value is 0 which implies calculation of all possible lines; increase it if you want to enhance the representation of EP;

- **n EP lines*eV/Å2** – the number of field lines calculated for eV/Å2.
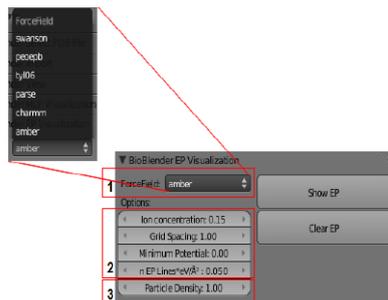
Now press **Show EP** button. *The process is time consuming as Show EP button invokes a custom software that calculates the field lines and exports them in the BioBlender 3D scene as NURBS curves. The positive end of each curve becomes an emitter. The particles flow along the curves from positive to negative.*

Change the **Particle Density** (3 in figure) to modify the number of the particles visualized in the scene. **Clear EP** to delete the curves and the emitters.

Raluca Mihaela Andrei1,2, Mike Pan1,* and Monica Zoppè1§

1 Scientific Visualization Unit, Institute of Clinical Physiology, CNR of Italy, Area della Ricerca, Pisa, Italy
2 Scuola Normale Superiore, Pisa, Italy
* Present address: University of British Columbia, Vancouver, Canada
§ Corresponding author
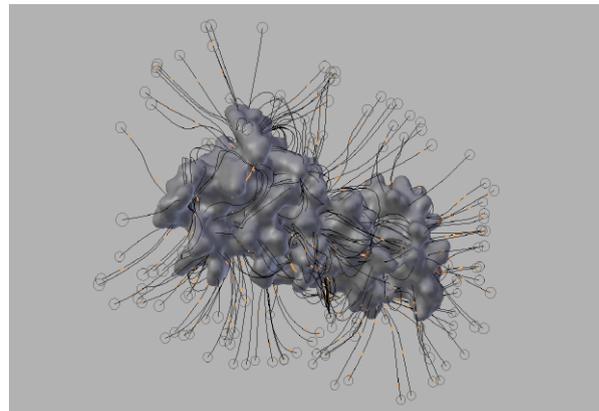
6. Output

To see the protein movement with the surface properties you have to render a movie. Since the movement implies a change of the atomic coordinates, the surface properties must be recalculated frame by frame.

In the **BioBlender Output** panel set the output file path (by default it is set to tmp folder); choose the kind of representation you prefer to render from the **Visualize** curtain menu:

- **Atom** – render only atoms;

- **Plain Surface** – render only surface;

- **MLP** – render surface with MLP;

- **EP + Plain Surface** – render surface (no MLP) and EP;

- **EP + MLP** – render surface with MLP and EP;

set **Start Frame** – the first frame of the animation;

set **End Frame** – the last frame of the animation;

set **Export Step** – the number of frames to skip during export, mostly used for faster export of .pdb files; enable **Information Overlay** to print extra information on the final image; enable Ambient Light only for GE visualization; do not enable it for MLP representation as its effect is confusing for MLP visual code.
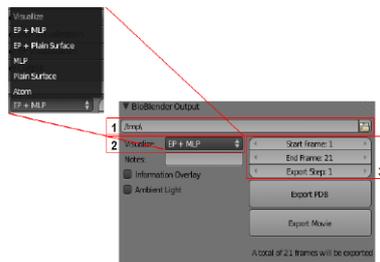
Hit **Export Movie** to render every frame of the animation. The output is a sequence of still images, this ensures that the rendering is resumed if the rendering process is disrupted. During section 3 Blender GE calculated and recorded intermediate conformations as keyframes. To save these coordinates as .pdb files for further analysis using external software, press **Export PDB**. A .pdb file is saved for each frame in the selected output.

To obtain the movie follow standard Blender procedures: open the Video Sequencer Editor: **Add -> Image**, select the sequence of images, go to **Properties** window and set the **Output** path and the **File Format** to **AVI JPEG** in the **Output** panel and **Start** and **End** frame in the **Dimensions** panel. Now press **Animation** button in the **Render** panel.
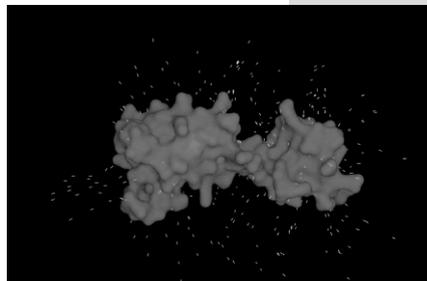
Now you have your protein moving with the surface properties visualized. An image of CaM with EP and MLP is shown in the image below ■

References

1 Zoppè, M; Porozov, Y; Andrei, R M; Cianchetta, S; Zini, M F; Loni, T; Caudai, C; Callieri, M (2008) Using Blender for molecular animation and scientific representation. *Proceedings* of the Blender Conference *Blender*

2 DeLano, WL, The PyMOL Molecular Graphics System, 2002

3 The PyMOL Molecular Graphics System, Version 1.2r3pre, Schrödinger, LLC

4 *Python*

5 *NumPy*

6 Kuboniwa H, Tjandra N, Grzesiek S, Ren H, Klee C B, Bax A (1995) Solution structure of calcium-free calmodulin. Nat Struct Biol 2: 768-76

7 Berman, H M; Westbrook, J; Feng, Z; Gilliland, G; Bhat, T N; Weissig, H; Shindyalov, I N; Bourne, P E (2000) The Protein Data Bank. Nucleic Acids Res 28: 235-42

8 Connolly, M L (1983) Solvent-accessible surfaces of proteins and nucleic acids. Science 221: 709-13

9 Zini, M F; Porozov, Y; Andrei, R M; Loni, T; Caudai, C; Zoppè, M (2010) Fast and Efficient All Atom Morphing of Proteins Using a Game Engine. (under review)

10 Testa, B; Carrupt, P A; Gaillard, P; Billois, F; Weber, P (1996) Lipophilicity in molecular modeling. Pharm Res 13: 335-43

11 Andrei R M, Callieri M, Zini M F, Loni T, Maraziti G, Pan M C, Zoppè, M (2010) A New Visual Code for Intuitive Representation of Surface Properties of Biomolecules. (under review) Raluca Mihaela Andrei1,2, Mike Pan1,* and Monica Zoppè1§

12 Scientific Visualization Unit, Institute of Clinical Physiology, CNR of Italy, Area della Ricerca,

13 Pisa, Italy

14 2Scuola Normale Superiore, Pisa, Italy

15 *Present address: University of British Columbia, Vancouver, Canada Corresponding author

Raluca Mihaela Andrei1,2, Mike Pan1,* and Monica Zoppè1§

1 Scientific Visualization Unit, Institute of Clinical Physiology, CNR of Italy, Area della Ricerca, Pisa, Italy
2 Scuola Normale Superiore, Pisa, Italy
* Present address: University of British Columbia, Vancouver, Canada
§ Corresponding author